PC- BASED REAL TIME SONIFICATION OF HUMAN MOTION CAPTURED BY INERTIAL SENSORS

Hans-Peter Brückner, Christopher Bartels, and Holger Blume

Leibniz Universität Hannover,
Institute of Microelectronic Systems,
Appelstraße 4,
30167 Hannover, Germany
{brueckner, bartels, blume}@ims.uni-hannover.de

ABSTRACT

This paper presents a low latency system for real time sonification of human motion captured by inertial sensors. Exemplarily the position of the wrist estimated by two inertial sensors located at upper arm and forearm is transformed into a continuous synthetic sound. The body segment orientation is captured by Xsens MTx sensors and used to compute the position of the wrist relative to the shoulder joint. The accessible motion parameters provided by inertial sensors are three axial segment orientations, three axial accelerations and angular rates, and all derived quantities like position in three dimensional space. Motion data sonification is performed by the Sound Synthesis Toolkit (STK), a set of C++ classes for audio signal processing and sound synthesis. The proposed framework enables future research in continuous real time sonification of human motion to improve the process of motion learning during stroke rehabilitation. Through software profiling the proposed framework is benchmarked in terms of latency induced by signal transmission and processing to evaluate maximal process-able sampling rates and inertial sensor counts.

1. INTRODUCTION

After a stroke the most of the patients are suffering from dramatically constrained mobility and partial paralysis of the extremities. Especially motion of the upper extremities, like grasping movements, is hampered. Grasping movements of stroke patients are typically characterized by less accuracy and a high variability. Additionally the movements are much slower and even simple tasks like grasping a water glass are very time consuming. In a rehabilitation sessions, the improvement is achieved by a large number of repetitions of the trained task supervised by a therapist. This kind of therapy requires a skilled therapist and ongoing training sessions.

The goal of stroke rehabilitation is to regain a maximum level of independence within daily activity. Therefore, many rehabilitation exercises focus on upper extremities movement, as these are base of tasks like eating, drinking and tooth brushing. Movements like grasping a water glass are mainly focused. Therefore, the inertial sensors have to be placed appropriate to capture these movements. In this approach two inertial sensors, attached to upper arm and forearm, allow computing the wrist position based on orientation data provided by two inertial sensors. The sonification acoustically displays the position of the wrist and allows the patient to detect if there is a continuous and target-orientated movement ideally without tremor. Several studies in the field of sports science show that motion learning benefits form motion sonification also efficacy in stroke rehabilitation is proved. Sonification in general is the displaying of non-speech information through audio signals [1].

This paper presents an approach for real time sonification of complex movements captured by inertial sensors. The system consists of a PC and an Xsens inertial sensor system [2] with up to ten MTx sensors and an Xbus Master device, performance is evaluated by a detailed profiling. The system provides flexibility to scale the number of MTx sensors according to motion capturing demands. The generated stereo audio signal is transmitted by speakers. The system components and structure is shown in Figure 1. Sensor data acquisition, sonification parameter calculation, and audio synthesis are handled on a PC with the attached motion capture system. As an example Sonification displays the wrist position in relation to the patient's body taking physical expectations into account.

The Sound Synthesis Toolkit (STK) is used for the sonification of the computed motion parameters [3]. The toolkit allows a sample based sound generation with a wide variety of configurable sound generators. The STK consists of audio signal processing and synthesis classes in C++. Thus, it allows seamless integration in the C++ based sensor system application programming interface (API) and orientation data processing framework. This first approach is based on the STK bowed instrument class for sound synthesis to estimate computational requirements of a complex sound generator.

For demonstration of the proposed system the calculated spherical coordinates of the wrist joint are mapped to frequency and left / right channel amplitude of the generated audio signal.

Another important parameter for the usage of inertial sensors in rehabilitation sessions is the long term drift of the sensor systems. The orientation and position drift of the captured data should not affect the generated sound signal while being in a static pose or vary over time when recapturing poses.

The paper is organized as follows: section 2 presents related work. Section 3 introduces inertial sensors and existing sonification frameworks for PC-based sound synthesis. The chosen motion parameter to sound synthesis mapping is presented in section 4. Section 5 illustrates the software structure and functional signal processing tasks. Detailed profiling results are given in section 6. A summary and an outlook to future work are presented in section 7.

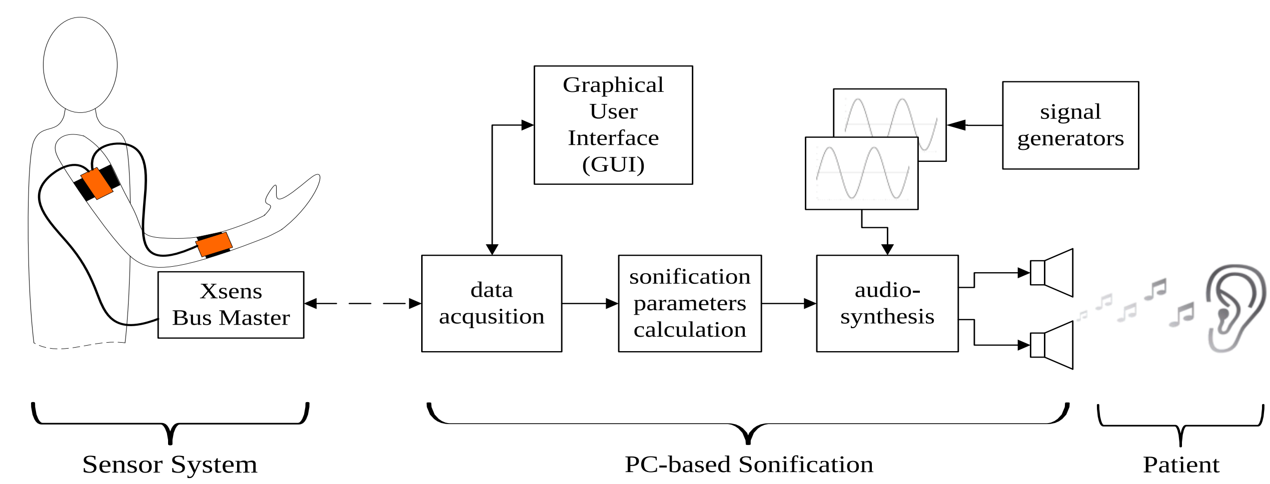

Figure 1: Structure of the proposed sonification system

2. RELATED WORK

Inertial sensors are a common approach for capturing movement data. There is a wide field of applications ranging from motion tracking for virtual reality applications to usage in rehabilitation [4], [5]. Inertial sensors overcome the lack of a permanent line of sight between the tracked object and the camera. This enables the mobile use of these sensors as the movement is not limited to a predefined area. Source less operation is another benefit of this sensing technique. Before starting a measurement session only a calibration step is required to align body and earth coordinate systems. Additionally, the single sensors are lightweight, have relative small dimensions compared to a camera based system and operate with a limited power budget.

Sonification based on metronome like stimuli in rehabilitation therapy of stroke patients showed significant improvements in the control of arm movements [6]. But this approach lacks a missing feedback and influence of the patient's motion on the generated audio stimuli. For a medically more effective sonification of motion data in the context of stroke rehabilitation the hardware has to perform a real time transformation of the captured motion parameters into a continuous synthetic audio signal. This enables immediate response based on movement information gathered by the audio signal.

Motion learning can be enhanced by continuous sonification of motion parameters. In [7] the visual presentation and convergent sonification of jumping movements, enhanced the precision and perception of subjects copying the prescribed sports movement compared to visual presentation only. Sonification is here mapped to ground reaction force caused by the demonstrator's jumps. In contrast to the proposed framework there is no real-time feedback, as the sonification bases on prerecorded jumping movements.

In [8] an acoustic biofeedback system for sonification of trunk kinematics is presented. The approach relies on a single inertial sensor attached at the trunk for movement capture. Three axis accelerations are mapped to stereo signal for direct feedback to improve balance while standing upright. The audio signal is manipulated in amplitude and frequency. Reductions in motion for keeping balance while using real time acoustic feedback are shown. This approach lacks a detailed analysis of latency due to signal processing and audio synthesis.

Research in [9] is also based on inertial sensors used for real time upper body motion capture. Measured body segment orientations of the five segment body model represent upper limb motion in the reachable sphere. Sphere positions are mapped to predefined sound samples. Sound samples are triggered by directional movements crossing predefined spatial segments. So motion within the sphere causes acoustic feedback.

A framework for sonification and visual feedback of reach and grasp tasks in stroke rehabilitation is presented in [10]. The motion capturing is based on a marker based camera system. Extracted movement features are used to generate visual and audio feedback in real time. Sonification uses various instruments and note values to indicate smoothness of the arm motion. Camera based motion tracking limits the usability of this approach to inpatient treatment, as a cost expensive and calibrated camera system is required. In contrast the system proposed here is applicable in home based rehabilitation.

Research in [11] also focuses on sonification design for rehabilitation purposes. Camera based motion capture allows the mapping of motion features like absolute height and velocity onto predefined sounds. The use of additional sonification in the exercises showed improvements in shoulder mobility. In contrast to the work proposed here the use of a limited number of predefined sounds does not allow a low latency representation of movements. Also the angular resolution is restricted to the limited number of sound files. Additionally, the camera based motion capture limits the usage to in patient treatment.

Summing it up, the proposed sonification framework overcomes existing frameworks for continuous sonification of arm movement in the lack of mobile usage, as most of them base on cameras for motion capture. In contrast inertial sensors provide source less and mobile operation. Furthermore, detailed software profiling in terms of latency is used to evaluate real-time capability. Orientation data mapping is based on a spherical coordinate system to provide intuitive usage.

3. SONIFICATION FRAMEWORK

In this section the general structure and components of inertial sensors and the used Xsens system will be introduced. Also an analysis of existing frameworks for PC-based sonification will be given to show possible frameworks for real time interactive sonification and their usability within the proposed system design.

3.1. Inertial Sensors

Inertial sensors consist of orthogonal mounted triads of gyroscopes, accelerometers and magnetometers. An additional temperature sensor is incorporated in some devices to compensate the sensors for temperature drift.

The sensor orientation in 3D space is then estimated by a sensor fusion algorithm. Possible fusion algorithms base on Kalman or Particle Filters, as described in [12], [13]. The Xsens MTx sensors use a Kalman Filter based sensor fusion algorithm, performed on a Digital Signal Processor (DSP) embedded in the sensor housing [2]. The orientation data output supports several modes, such as unit quaternions, euler angles and rotation matrix. In this work the rotation matrix mode is used.

The achievable maximum sampling frequency depends on the number of motion trackers connected to the Xbus Master device. A single sensor can be sampled with a maximum frequency of 120 Hz if the data output is set to orientation data. In contrast to orientation data the raw data of the accelerometer, magnetometer and gyroscope triads can be sampled up to 512 Hz. Using this raw data output mode sensor fusion has to be performed by a user defined algorithm on the user's hardware platform. The sensors are able to capture movement with an angular resolution of 0.05° and a static accuracy of 1.0°. During movement the accuracy is 2.0°. A single Mtx sensors size is 38 mm by 53 mm by 21 mm, and has a weight of 30 gram.

To capture the patients arm movement two MTx sensors are mounted according to Figure 2. One sensor is placed on the forearm and the other on the upper arm. Both sensors are fixed using Velcro straps. Several sensors are connected in series to the Xbus Master device. The shown wide spread arm pose is used for the initial calibration of the sensors. The figure applies to left arm sonification. The coordinate system x axis is then aligned to the arm, the y axis pointing to the front and z axis aligned to the vertical body axis. The pose can be adopted while standing upright or sitting up straight.

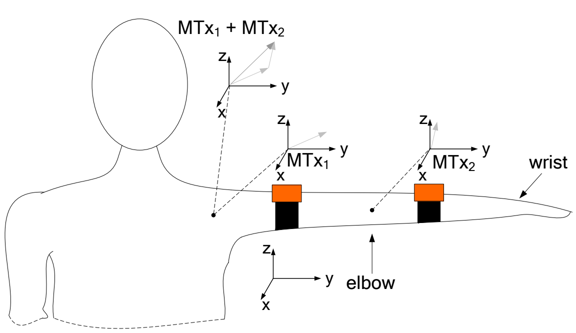

Figure 2: Placement of the MTx inertial sensors

3.2. Sound Synthesis Environments

Real time sonification of the captured movement features requires the real time modulation and synthesis of a generated audio signal. Therefore, the audio buffer has to be changed according at least to the sensor system sample frequency. Assuming a 100 Hz sampling frequency this would result in a maximum audio buffer size of 10 ms.

Additionally, to real time sonification the ideal environment has to be C / C++ based and platform independent, to enable future usage on hardware platforms like combinations of RISC and DSP or RISC and FPGA. Another criterion is the direct access of the software to the audio buffer, because file based read and write operations would increase the system latency and require an existing file system.

As the sensor communication is based on a C / C++ API the sound synthesis framework should also provide such an API or source code. Table 1 lists some established sonification environments and their programming language and usability in a C++ based software environment. The table additionally shows the type of license and the framework programming language. The source code language of all frameworks listed in Table 1 is C or C++, so no additional coding language skills are required for custom changes in predefined instruments or other framework parts. However, there are a variety of sonification environments, some based on Java [14] and other programming languages [15], [16].

Name	License	Programming language	C / C++ API
ChucK [17]	GPL	C++	No
Pure Data [18]	SIBSD	C	No
SuperCollider [19]	GPL	C / C++	No
Max/MSP [20]	Licensed	C	No
Nsound [21]	GPL	C++	Yes
Csound [22]	LGPL	C / C++	Yes
Sound Synthesis Toolkit (STK) [3]	Free	C / C++	Yes

Table 1: Sonification environments

3.2.1. Synthesis Frameworks without C / C++ Interface

Chuck [17] is a standalone audio programming language for real time audio synthesis, but it does not provide any C / C++ interface. Pure Data [18] is a graphical real time audio programming language. It can be extended by new sound objects written in C or C++, but does not provide an API. The high level audio synthesis language SuperCollider [19] is also capable of real time audio synthesis. It allows the control of MIDI devices. Max/MSP [20] is a charged graphical environment for audio signal processing and synthesis. Through C based extensions allow the definition of custom generators.

As mentioned before the focus in this application was seamless integration of an audio synthesis environment into a C++ based sensor communication and data processing framework. Therefore, these environments are disregarded in the further conceptual and designing phases.

3.2.2. Synthesis Frameworks with C / C++ API

Nsound [21] is a C++ and Python library for computer-based audio synthesis. Real time sound synthesis is not capable. Thus this environment is not capable of audio synthesis tasks required in the proposed project.

Csound [22] and STK [3] both provide real time sound synthesis. Both of them are expandable with custom sound generators written in C or C++. However, in this project context the file based synthesis of Csound, requiring an orchestra and score file for sound synthesis increases latency, compared to a direct sample based synthesis provided by STK. Independence from file based operation is an important requirement for future portability of the software to RISC / DSP based hardware platforms for mobile use in rehabilitation exercises.

STK provides large flexibility as the framework consists of processing, generation and output classes, capable of being integrated in every custom C / C++ software. Sound synthesis and audio signal processing functionality is accessible through open source C++ based functions and classes. Audio processing is sample based whereby audio buffer is computed by STK signal generation classes. Through specialized or customized generators the synthetic sound can be adapted to user requirements. For sonification of the inertial sensors in the proposed framework the STK bowed instrument class operating at distinct scale frequencies is used. The instrument provides three degrees of freedom – amplitude, frequency and panning. A sonification of additional parameters and characteristics could be achieved by mixing different musical instruments.

4. MAPPING: POSITION DATA TO SOUND

Mapping the wrist position to audio samples generated by a STK sound generator class, called Bowed instrument, allows interactive sonification, as motions are directly transformed into sound. Representation of three dimensional coordinates by an acoustic signal requires three degrees of freedom within the synthesized audio signal. Using sound generators this could be amplitude, frequency and panning. To provide a sonification, meeting physical expectations the proposed system uses a spherical coordinate system.

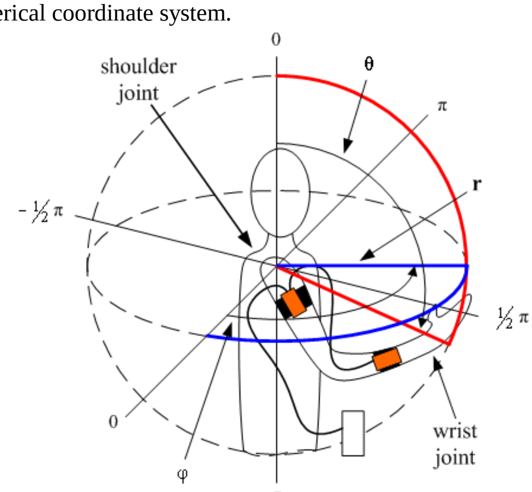

Figure 3: Spherical coordinate system conventions

The spherical coordinate systems origin is located at the shoulder joint according to Figure 3. Thus, radius *r* represents the shoulder joint to wrist joint distance. Azimuth angle φ represents the horizontal wrist position, polar angle θ the vertical displacement. Spherical coordinate system is chosen as own experiments show, that it provides a more intuitive sonification compared to a Cartesian coordinate system.

The computed radius is normalized corresponding to the user’s upper arm and forearm length, being specified on the GUI before starting sonification. This ensures independence of maximal loudness form body height to avoid harm caused by operating close to or beyond pain threshold.

The radius influences the amplitude *A* of the synthesized audio signal. According to physical laws the sonification gets louder while the arm is moving towards the patient and more quiet during departing. This behavior is linear modeled according to (1). Slope parameter was experimentally defined regarding to the normalized reachable sphere. The offset ensures a low noise in case of a totally extended arm.

$$A = \left(2 - \frac{9}{5} * r\right) \tag{1}$$

Panning or panorama of the audio signal is controlled by the azimuth angle φ. Left and right channel amplitude is therefore calculated according to equation (2) respectively (3). Normalization of the coordinate system ensures operating with audio signal loudness without impairment of health. Changing loudness of let and right audio channel indicates whether the arm is on the left or right of the patient.

$$\text{left channel volume} = A * \left(\varphi - \frac{1}{3}\pi\right) \tag{2}$$

$$\text{right channel volume} = A * \left(\frac{2}{3}\pi - \varphi\right) \tag{3}$$

The STK bowed sound generators frequency is adapted according to the polar angle θ, as shown in Table 2. Angles ranging form ¼ π to ¾ π are equidistantly mapped to the notes a to as’’. Polar angle values above 2.42 radians are mapped to the note a and angle values below 0.79 radian are mapped to as’’. This results in an angular difference between two notes of approximately $\frac{1}{44}\pi$, as angular resolution for wrist joint is 2.0 deg according to [23].

Note	Polar angle θ	Note	Polar angle θ
a	> 2.42	a'	1.57 - 1.64
b	2.36 - 2.42	b'	1.50 -1.57
h	2.28 - 2.36	h'	1.43 - 1.50
c'	2.21 - 2.28	c"	1.35 - 1.43
des'	2.14 - 2.21	des"	1.28 - 1.35
d'	2.07 - 2.14	d"	1.21 - 1.28
es'	2.00 - 2.07	es"	1.14 - 1.21
e'	1.92 - 2.00	e"	1.07 - 1.14
f'	1.86 - 1.92	f"	1.00 - 1.07
ges'	1.78 - 1.86	ges"	0.93 - 1.00
g'	1.71 - 1.78	g"	0.86 - 0.93
as'	1.64 - 1.71	as"	< 0.79

Table 2: Polar angle to sound generator frequency mapping

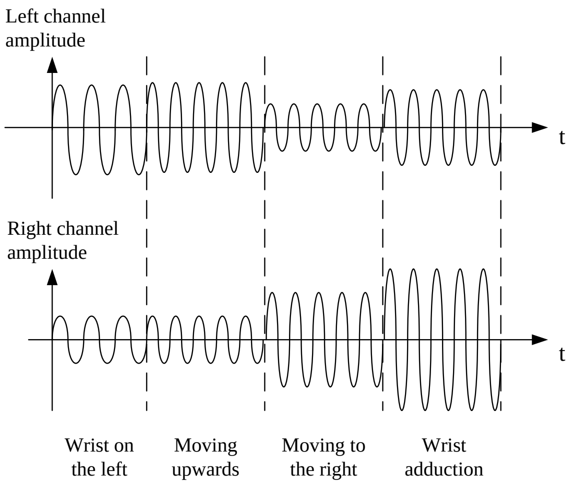

Figure 4: Wrist position influence on time domain signal

Figure 4 shows wrist position influence on the generated audio signal in the time domain. The figure illustrates the amplitude and frequency changes during occupying static poses. The signal amplitude presented variation in time therefore only clarifies the principle and does not correspond to a moving arm, as transitions are disregarded. Initially the wrist is located somewhere on the left, as the left channel amplitude is greater than the right. Moving the wrist upwards results in an increased frequency, as shown in the second section. A movement to the left on equal height changes the amplitude ratio only, represented in section 3. Section 4 depicts moving the wrist towards oneself. This results in increased amplitudes on both channels at a constant amplitude ratio.

5. SOFTWARE DESIGN

The proposed software for sonification of inertial sensor data is based on the producer consumer pattern to parallelize communication to the Xsens sensor system, computation of movement features and sonification. Furthermore, it allows a decoupling of communication and data processing functionality working at different data rates. A particular requirement is a flexible sound generator to allow adopting medical effective sonification algorithms in future.

Figure 5 shows the structure of the designed software. The proposed application consists of four main software classes named SonSensQt, XsensData, HandleData and Sound. For controlling sonification, displaying sensor connection status and actual wrist position a graphical user interface (GUI) was designed. The user interface is Qt based, a cross platform framework specialized on graphical user interfaces [24].

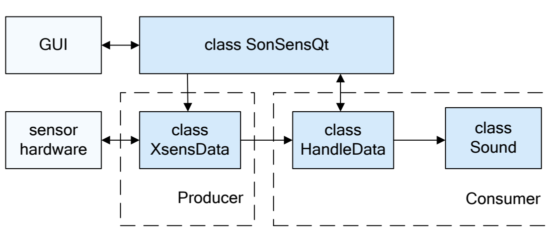

Figure 5: Software design

The SonSensQt class handles the user events from the GUI and reformats the data for displaying on the GUI. The Qt signals & slots concept is used for communication between the class objects. The concept ensures type safe connection of the objects with flexibility needed for later extension of the application. The configuration of the sensor system and data item enqueue is managed by the XsensData class. Main custom options are the sampling frequencies of the attached motion trackers and different orientation data output formats. The class also provides reset functionality of the attached motion trackers for calibration purposes for coordinate system alignment. The HandleData class dequeues data items and computes movement features, like position, for sonification. The Sound class handles audio output by implementing the corresponding STK functions. Encapsulation of the sonification functionality enables future extensions by adding new sound generators or mappings without changing the parameter computation flow.

Data exchange between the classes XsensData and HandleData is handled via a buffer for the captured inertial sensor data. When receiving new sensor data the Xsens data structure is converted to a static data structure containing the rotation matricx. Synchronization between the threads is achieved by a mutual exclusion (mutex) element.

For audio output the STK RtAudio class is used. This class provides functionality to open a real time audio input and output stream. Using the RtAudio openStream() function a two channel audio with a 44.1 kHz sample rate connected to the default audio device is created. In this case the development computers sound card. Parameters are adapted to achieve real-time performance of the generated audio stream and minimize buffer latency by choosing minimum buffer size of 441 samples per channel

The data and control flow after presetting required parameters like initializing the sensor system, configuration of the arm length is shown in Figure 6. The sensor system is polled by the producer thread implemented in the XsensData class. Captured new data items are inserted in the buffer making the data available for further processing. After removing a data element form the buffer the movement features to be displayed through sonification are calculated. For demonstration the wrist position is computed out of the rotation matrices representing the orientation of upper arm and forearm. Wrist position and sensor orientation matrices are displayed on the GUI. The Cartesian position is transformed into spherical coordinates for sonification. The Sound class implements a callback function, which is called to compute the audio buffer. To synchronize between operating system audio buffer and updating sonification parameters a mutex is used to avoid concurrent access to the parameters frequency and panning (left / right amplitude) used to compute the actual audio buffer. Adapting the sonification sound to user needs and preferences can be achieved by mixing changing STK the sound generators used. Additional sound generators could also be used to display additional motion information like elbow position or motion velocity and acceleration.

The data flow shown in Figure 6 will also be used in section 6 to determine the overall system latency, as it shows the applications critical path. Therefore, the functional block and their sub functions are analyzed to show major influences on real-time performance of the system.

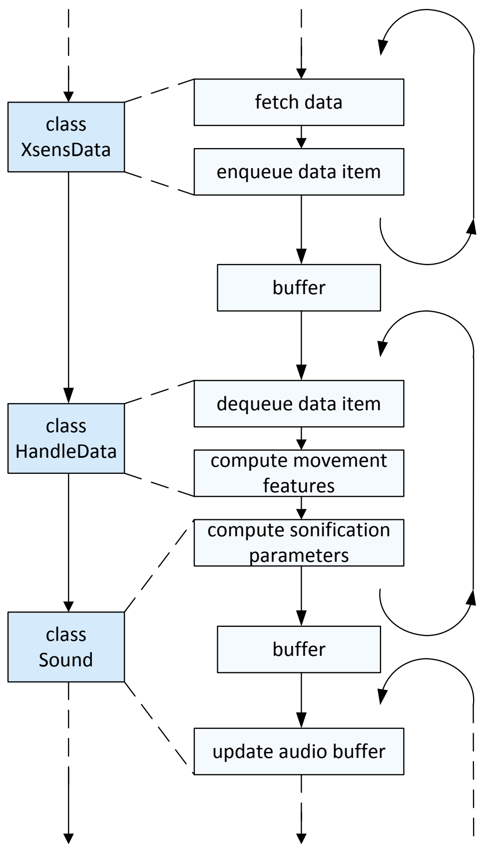

Figure 6: Data and control flow

6. PROFILING RESULTS

Overall system latency can be sub-divided into three blocks. The first fraction is the data acquisition time of the sensors and the transmission time from sensor system to the host PC. The second part includes all latency induced by computations on the PC. The last part consists of the delay caused by the necessary audio buffer size, when using the Microsoft DirectSound API.

For software profiling the instrumentation method of the Visual Studio 2010 Ultimate Profiling Tool is used. The instrumentation method provides detailed runtime data of every function including external calls. Elapsed inclusive time values presented here show the time spent in the individual function and sub functions including time spend in calls to the operation system like context switches or input and output operations. Using these values for benchmarking is required as audio output is heavily depending on output operations. Application inclusive values, only incorporating time spend in the function and its sub functions would hide these input / output operations induced latency. The development PC, also used for latency analysis, is equipped with an Intel Core2Duo E8400 CPU @ 3 GHz and 3 GB RAM.

The transmission and data acquisition time is calculated according to the Xsens XM-B user manual. Orientation matrix output mode requires transmitting 36 bytes per sample for each attached MTx motion tracker. Additionally, the message structure, according to Figure 7, contains 7 bytes. The preamble, bus identifier, message identifier, length, checksum and sample counter. Thus, the proposed configuration with two MTx sensors in orientation matrix mode requires a transmission of 79 bytes for each sampling period. The given example describes the structure of a data message for transmitting the data in orientation matrix format and a sample counter in the case of two attached MTx sensors. The transmission time per sampling period for a communication using one stop bit is calculated according to (4).

$$transmission\ time = \frac{message\ bytes * 9\ (\frac{bit}{byte})}{communication\ baudrate\ (\frac{bit}{s})} \tag{4}$$

The system uses a baud rate of 115200 bps. This results in a transmission induced latency of 6.17 ms. Data acquisition time and orientation data calculation of each MTx sensor ranges from 0.31 ms to 2.55 ms as the calculations of the Xsens Kalman Filter depend on the measurements. Therefore, for latency calculation the worst case of 2.55 ms acquisition and calculation time is assumed. Resulting acquisition and data transmission time of the proposed system then is 8.72 ms.

The software caused latency is divided in the functional blocks for data processing shown, in Figure 6. Values listed in the next paragraph indicate the time needed for each task to compute an update of the sonification parameters. For example this contains enqueue of two orientation matrix items.

Fetching and analyzing data items takes 0.67 µs. Enqueue for two data items costs 0.76 µs, the dequeue process takes 0.77 µs. Computation of the spherical wrist position based on the rotation matrices for upper and forearm lasts 1.46 µs. Formatting computed data for displaying on the GUI and emit of the corresponding signals takes 51.50 µs. It takes 63.42 µs to set the audio signal parameters, frequency and panning, and computing the audio buffer is. Cumulative computational latency is 118.58 µs. Figure 8 shows the percentage distribution of the aforementioned computational tasks. It shows that main time consuming computational tasks are audio buffer computation and displaying the movement parameters.

The central latency element is the minimum required audio buffer size of at least 441 samples per channel, as otherwise glitches appear in the generated audio signal. Using the STK RtAudio class and the Microsoft DirectSound API enforce a buffered access to the PC systems audio device. Thus, the inherent latency induced through audio buffer size is 10 ms. Porting the sonification framework to a hardware platform providing a direct access to an on board audio codec chip would allow unbuffered access to the audio output device and thus result in significantly reduced system latency.

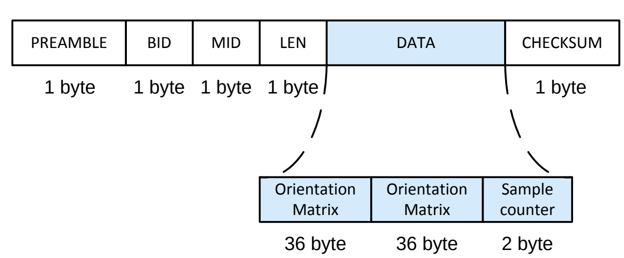

Figure 7: Xsens message structure

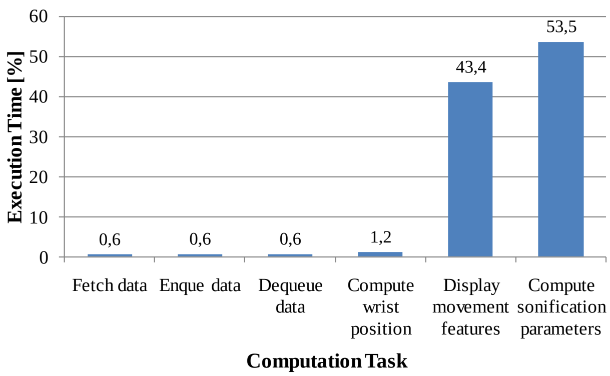

Figure 8: Computational latency percentage distribution

Overall latency of the proposed system comprises of data acquisition, computational and audio buffer latency and results in an entire latency of 18.84 ms. Table 3 presents the latency in ms induced by the main framework blocks acquisition and transmission, computation of movement features and the necessary audio buffer. Furthermore the percentage latency for each block is given.

Data throughput of the proposed framework allows operating at 100 Hz, the maximum sampling frequency of the attached MTx sensor system with two sensors attached. Pipelined computation, as shown in Figure 6 limits the maximum computational time in a stage to 63.42 µs, the time required for audio buffer computation.

System sub-block	Latency [ms]	Latency [%]
Acquisition and transmission	8.72	46.28
Computation	$118.58*10^{-3}$	0.63
Audio buffer	10.00	53.08

Table 3: Latency by system component

7. CONCLUSION

The general software design goal was to achieve a continuous synthetic upper arm movement sonification for application in stroke rehabilitation. Motion capturing is based on an inertial sensor system, as this kind of sensor enables mobile usage. The system latency constraint was 30 ms, as higher values result in latency significantly recognizable for humans [25]. An overall latency of 18.84 ms of the proposed continuous sonification approach meets this demand and contains margin for further development.

Profiling results show that overall system latency is mainly caused by essential audio buffering, as there is no direct access to the audio buffer on a PC-based platform. The second central part is the inherent MTx sensor data acquisition and data transmission time. These two tasks cause about 99 percent latency. As sensor system latency is inherent, further latency reductions have to focus on minimizing the audio buffer size or changing the data format to be transmitted to quaternion, as this change would nearly half the message size and thus the transmission latency.

Profiling results clarify that more complex audio signal generation including mixing different fundamental or instrumental sound generators blocks would not significantly increase total latency. Per added sound generator the latency would increase approximately by 63.42 µs, according to the given profiling results. This enables further research in designing more comfortable and medical effective parameter mappings for audio synthesis, by designing a more complex audio synthesis. One way could be the mapping of additional motion information to different sound synthesis blocks to enable their concurrent sonification. Elbow position and body segment velocity or accelerations could be complementally, useful motion parameters, as they also have been used in other motion sonification approaches. Also the design of a more pleasant audio generator requires further research.

In summary the proposed system enables real-time low latency sonification, provides the required flexibility for adoptions in movement feature calculations and sound synthesis, and enables further research in sonification design for upper arm movements. The system will also be used in studies to determine benefit from a continuous synthetic sonification in reach and grasp motor learning tasks. Studies will be used to figure out significant motion parameters for relearning of movements and the design of an effective parameter to sound mapping as well as an ambient and motivating sound design. It is a research platform for further audio design and movement feature calculations for designing a more effective and pleasant sonification for use in home based stroke rehabilitation.

Future work will be porting the proposed system to low-power hardware platforms, to enable mobile usage. Research therefore focuses on exploring optimized hardware architectures for sonification of inertial sensor data, to enable the mobile use of the system. The proposed software allows choosing an optimized hardware platform by analyzing computational cost and latency for each individual function through detailed software profiling. Combined with wireless sensor connectivity the system could be used in home based stroke rehabilitation. This approach could also result in a drastically reduced audio buffer size, as alternative hardware platforms allow direct audio buffer access. This makes large buffer sizes needless. In addition overall system latency would directly benefit from the mentioned hardware architecture changes.

8. ADDITIONAL FILES

The attached “motion2sound.wav” file represents an arm moving from the right to the front, then grasping a cup, moving it to the left and back to front. After that the cup is raised for drinking and put back on to the table on the right. The file is available for download at http://www.ims.uni-hannover.de/fileadmin/www/files/forschung/sonification/motion2sound.wav .

9. ACKNOWLEDGMENT

The work for this research project (W2-80118660) has been financially supported by the “Europäischer Fonds für regionale Entwicklung” (EFRE).

10. REFERNCES

[1] G. Kramer, *An introduction to auditory display.*: Addison Wesley Longman, 1992.

[2] Xsens Technologies BV. [Online]. www.xsens.com

[3] G.P. Scavone and P.R. Cook, "RtMidi, RtAudio, and a synthesis toolkit (STK) update," *In Proceedings of the International Computer Music Conference*, 2005.

[4] Y. Tao and H. Hu, "3D arm motion tracking for home-based rehabilitation," *Proceedings of the 3rd Cambridge Workshop on Universal Access and Assistive Technology*, pp. 10-12, 2006.

[5] Eric R. Bachmann, Robert B. McGhee, and Xiaoping and Zyda Yun, "Inertial and magnetic posture tracking for inserting humans into," *Proceedings of the ACM symposium on Virtual reality software*, pp. 9-16, 2001.

[6] E. Taub et al., "Technique to improve chronic motor deficit after stroke," *Archives of physical medicine and rehabilitation*, pp. 347-354, 1993.

[7] A. Effenberg, "Movement Sonification: Motion Perception, behavioral effects and functional data," *Proceedings of the 2nd International Workshop on Interactive Sonification*, 2007.

[8] D. Cappello, Giansanti M., Dozza L., and Chiari G. and Maccioni A., "Energetic assessment of trunk postural modifications induced by a wearable audio-biofeedback system," *Medical Engineering & Physics*, pp. 48-54, 2009.

[9] P.J. Maes, M. Leman, and M. Lesaffre, "A model-based sonification system for directional movement behavior," *Interactive Sonification Workshop*, 2010.

[10] I. Wallis et al., "Real-Time Sonification of Movement for an Immersive Stroke Rehabilitation Environment," *Proceedings of the 13th International Conference on Auditory Display*, 2007.

[11] K. Vogt, D. Pirrò, I. Kobenz, R. Höldrich, and G. Eckel, "PhysioSonic-Evaluated Movement Sonification as Auditory Feedback in Physiotherapy," *Auditory Display*, pp. 103-120, 2010.

[12] D. Roetenberg, H. Luinge, C. Baten, and P. Veltink, "Compensation of magnetic disturbances improves inertial and magnetic sensing of human body segment orientation," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, pp. 395 -405, 2005.

[13] Won Seong-hoon, W. Melek, and F. Golnaraghi, "Position and orientation estimation using Kalman filtering and particle," *Industrial Electronics*, pp. 3006-3011, 2008.

[14] K. van den Doel and D.K. Pai, "JASS: A Java audio synthesis system for programmers," *Proceedings of the International Conference on Auditory Display 2001*, pp. 150-154, 2001.

[15] R. Dannenberg and E. Brandt, "A flexible real-time software synthesis system," *Proceedings of the 1996 International Computer Music Conference*, pp. 270-273, 1996.

[16] A. Sorensen, "Impromptu: An interactive programming environment for composition and performance," *Proceedings of the Australasian Computer Music Conference 2009*, pp. 149-153, 2010.

[17] G. Wang and P.R. Cook, "ChucK: A concurrent, on-the-fly audio programming language," *Proceedings of International Computer Music Conference*, pp. 219-226, 2003.

[18] M. Puckette, "Pure Data: another integrated computer music environment," *Proceedings of the Second Intercollege Computer Music Concerts*, pp. 37-41, 1996.

[19] J. McCartney, "SuperCollider, a new real time synthesis language," *Proceedings of the International Computer Music Conference*, pp. 257-258, 1996.

[20] Cycling'74. Max/MSP: Interactive visual programming environment for music, audio and media. [Online]. http://cycling74.com/products/maxmspjitter/

[21] Nsound. Homepage of the Nsound project. [Online]. http://nsound.sourceforge.net/index.html

[22] CSound. CSound homepage. [Online]. http://www.csounds.com/about

[23] H. Tan, M. Srinivasan, B. Eberman, and B. Cheng, "Human factors for the design of force-reflecting haptic interfaces," *Dynamic Systems and Control*, pp. 353-359, 1994.

[24] Nokia Corporation. Qt - A cross-plattform application and UI framework. [Online]. http://qt.nokia.com/

[25] D. Levitin, K. MacLean, M. Mathews, L. Chu, and E. Jensen, "The perception of cross-modal simultaneity," *International Journal of Computing Anticipatory Systems*, 2000.